Efficient Techniques for Formal Verification of PowerPC 750 Executables, Phase II

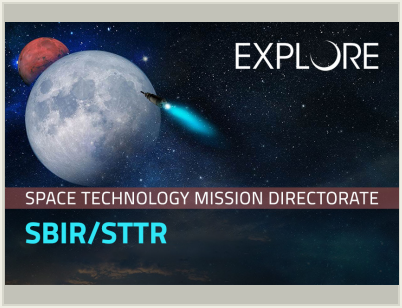
Completed Technology Project (2009 - 2014)

Project Introduction

We will develop an efficient tool for formal verification of PowerPC 750 executables. The PowerPC 750 architecture is used in the radiation-hardened RAD750 flight-control computers that are utilized in many space missions. The resulting tool will be capable of formally checking: 1) the equivalence of two instruction sequences; and 2) properties of a given instruction sequence. The tool will automatically introduce symbolic state for state variables that are not initialized and for external inputs. We bring a tremendous expertise in formal verification of complex microprocessors, formal definition of instruction semantics, and efficient translation of formulas from formal verification to Boolean Satisfiability (SAT). We will also produce formally verified definitions of the PowerPC 750 instructions used in the project, expressed in synthesizable Verilog; these definitions could be utilized for formal verification and testing of PowerPC 750 compatible processors, for FPGA-based emulation of PowerPC 750 executables, as well as in other formal verification tools to be implemented in the future.

Anticipated Benefits

The capability to automatically generate a symbolic simulator for an ISA, given a formal definition of its semantics, will dramatically increase the potential for commercialization of the proposed technology. All companies that either manufacture microprocessors or develop IP of microprocessors, as well as all their clients, will be potential users. As embedded microprocessors are increasingly used in safety critical applications, it will become the norm to formally verify the executables for such applications. The immediate non-NASA commercialization will target the members of Power.org, an organization whose purpose is to develop, enable, and promote PowerPC Architecture technology. Power.org has over 40 member companies. The PowerPC architecture is used in many safety-critical embedded systems. Non-NASA customers of this technology will similarly be able to use the tool to formally verify the equivalence of two instruction sequences, and to formally check properties for a given executable. Furthermore, non-NASA customers will be able to use the tool to detect security vulnerabilities in programs, thus ensuring their robustness to security attacks, as well as to detect malicious intent in executables. The last application will allow the technology to be used in sophisticated virus scanners, utilizing formal reasoning to ensure robustness to software obfuscations of malicious intent.



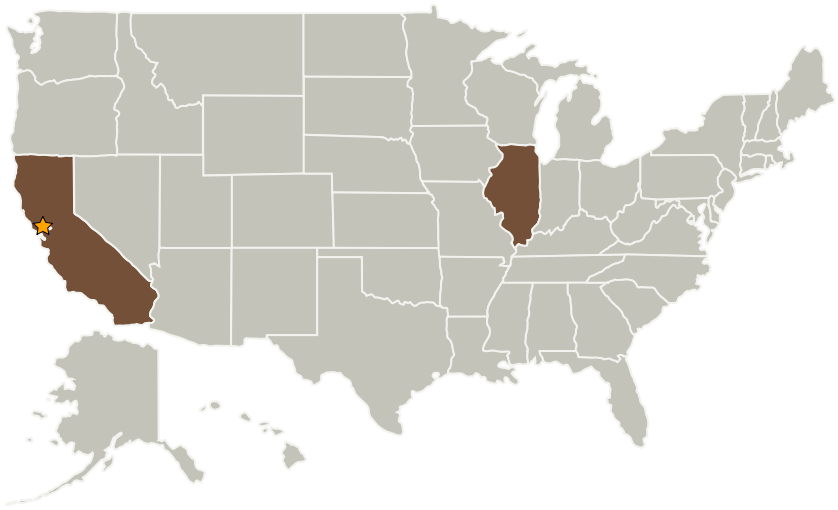
Efficient Techniques for Formal Verification of PowerPC 750 Executables, Phase II

Table of Contents

Efficient Techniques for Formal Verification of PowerPC 750 Executables, Phase II

Completed Technology Project (2009 - 2014)

Primary U.S. Work Locations and Key Partners



Organizations Performing Work	Role	Type	Location
★Ames Research Center(ARC)	Lead Organization	NASA Center	Moffett Field, California
Aries Design Automation, LLC	Supporting Organization	Industry	Chicago, Illinois

Co-Funding Partners	Type	Location
Self-Actualizer, LLC	Industry	

Primary U.S. Work Locations	
California	Illinois

Project Transitions

March 2009: Project Start

Organizational Responsibility

Responsible Mission Directorate:

Space Technology Mission Directorate (STMD)

Lead Center / Facility:

Ames Research Center (ARC)

Responsible Program:

Small Business Innovation Research/Small Business Tech Transfer

Project Management

Program Director:

Jason L Kessler

Program Manager:

Carlos Torrez

Project Manager:

Andre Goforth

Principal Investigator:

Miroslav N Velev

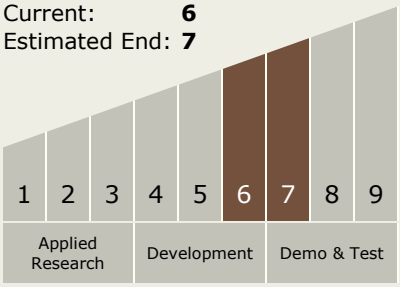
Efficient Techniques for Formal Verification of PowerPC 750 Executables, Phase II

Completed Technology Project (2009 - 2014)

May 2014: Closed out

Technology Maturity (TRL)

Start: **6**
Current: **6**
Estimated End: **7**



Technology Areas

Primary:

- TX11 Software, Modeling, Simulation, and Information Processing
  - TX11.1 Software Development, Engineering, and Integrity
    - TX11.1.7 Frameworks, Languages, Tools, and Standards